IMPLICIT THERMAL LOGIC IN FLAC
(Revised)

(Itasca Document Review 006-01-42)

Prepared for:

U.S. Nuclear Regulatory Commission
Division of Waste Management
under Contract NRC-02-85-002

Prepared by:

Itasca Consulting Group, Inc.
Suite 210
University Technology Center
1313 5th Street SE
Minneapolis, Minnesota 55414

October 1988

IMPLICIT THERMAL LOGIC IN FLAC

(Itasca Document Review 006-01-42)

Prepared for: U.S. Nuclear Regulatory Commission
Contract No. NRC-02-85-002
Task Order No. 006, Task 1

Task Description: to complete an implicit thermal algorithm to the computer code FLAC

Prepared by: Itasca Consulting Group, Inc.
Suite 210
1313 5th Street SE
Minneapolis, Minnesota 55414

Author:

MGMack

Mark G. Mack
Itasca Consulting Group, Inc.

Date: 10/28/88

Reviewer:

BBrady

Barry Brady
Itasca Consulting Group, Inc.

Date: 10/31/88

Approved:

Roger Hart

Roger Hart
Project Manager

Date: 10/31/88

TABLE OF CONTENTS

IMPLICIT THERMAL LOGIC IN FLAC

1.0 INTRODUCTION

An implicit thermal algorithm has been implemented in FLAC. The algorithm used is the Crank-Nicholson method, and the set of equations is solved by an iterative scheme known as the Jacobi method. An implicit method is advantageous for solving linear problems such as heat conduction with constant conductivity, because it allows the use of much larger timesteps than those permitted by an explicit method, particularly at later times in a problem, when temperatures are changing slowly.

2.0 THEORY

The usual one-dimensional explicit finite difference scheme for heat conduction can be written

$$\frac{\rho C_p}{k} \bullet \frac{T_i(t + \Delta t) - T_i(t)}{\Delta t} = \frac{T_{i+1}(t) - 2T_i(t) + T_{i-1}(t)}{(\Delta x)^2} \qquad (1)$$

An implicit method can be derived by replacing the right-hand side of Eq. (1) by the expression:

$$\frac{1}{2}\left[\frac{T_{i+1}(t+\Delta t) - 2T_i(t + \Delta t) + T_{i-1}(t + \Delta t)}{(\Delta x)^2} + \frac{T_{i-1}(t) - 2T_i(t) + T_{i-1}(t)}{(\Delta x)^2}\right]$$

This method, known as the Crank-Nicholson method, has the advantage that it is stable for all values of Δt, but has the disadvantage of being implicit. This means that the temperature change at any point depends on the temperature change at other points. (This can be seen by rewriting the implicit scheme as

$$\frac{\rho c p}{k\Delta t}\,\Delta T_i = \left[\frac{T_{i+1} + \frac{1}{2}\Delta T_{i+1} - 2(T_i + \frac{1}{2}\Delta T_i) + T_{i-1} + \frac{1}{2}\Delta T_{i-1}}{(\Delta x)^2}\right]$$

since $T_k(t + \Delta t) = T_k(t) + \Delta T_k$.)

The implicit method requires that a set of equations be solved at each timestep for the values of ΔT_i.

In matrix notation, the <u>explicit</u> method can be written as

$$\underline{\Delta T} = \underline{\underline{C}}\ \underline{T}$$

where $\underline{\underline{C}}$ is a coefficient matrix,

$\underline{T}$ is a vector of the temperatures, and

$\underline{\Delta T}$ is a vector of the temperature change.

The <u>implicit scheme</u> can be written as

$$\underline{\Delta T} = \underline{\underline{C}}(\underline{T} + \tfrac{1}{2}\underline{\Delta T})$$

which can be rewritten as

$$(\underline{\underline{I}} - \tfrac{1}{2}\underline{\underline{C}})\underline{\Delta T} = \underline{\underline{C}}\ \underline{T}$$

where we need to solve for $\underline{\Delta T}$ at each timestep.

The matrix

$$(\underline{\underline{I}} - \tfrac{1}{2}\,\underline{\underline{C}})$$

is diagonally dominant and sparse, because only neighboring points contribute non-zero values to $\underline{\underline{C}}$.

This set of equations is thus efficiently solved by an iterative scheme. For ease of implementation as a simple extension of the already available explicit method, the Jacobi method is used. For the NxN system $\underline{Ax}=\underline{b}$, this can be written for the n^{th} iteration as

$$x_i(n+1) = \frac{b_i}{a_{ii}} - \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\frac{a_{ij}}{a_{ii}}\, x_j(n) \right] \qquad \begin{array}{l} i = 1,2, \;. \;. \;. \;N \\ n = 1,2, \;. \;. \;. \end{array}$$

that is,

$$x_i(n+1) = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1}^{N} a_{ij}\, x_j(n) \right] + x_i(n)$$

In our case, this becomes,

$$\Delta T_i(n+1) = \frac{1}{(1-\frac{1}{2}C_{ii})} \left[\sum_{j=1}^{N} C_{ij}\, T_j - \sum_{j=1}^{N} (\delta_{ij} - \tfrac{1}{2}C_{ij})\Delta T_j(n) \right] + \Delta T_i(n)$$

$$= \frac{1}{(1-\frac{1}{2}C_{ii})} \left[\sum_{j=1}^{N} C_{ij}\, T_j + \frac{1}{2} \sum_{j=1}^{N} C_{ij}\, \Delta T_j(n) - \Delta T_i(n) \right] + \Delta T_i(n)$$

This equation shows the analogy between the implicit scheme and the explicit scheme which can be written as

$$\Delta T_i = \sum_{j=1}^{N} C_{ij} \, T_j$$

The amount of calculation required for each timestep is approximately n + 1 times that required for one timestep in the explicit scheme, where n is the number of iterations per timestep. This extra calculation can be more than offset by the much larger timestep permitted by the implicit method. However, the implicit scheme can give poor accuracy because it assumes that the temperature change is a linear function of time in a single timestep, which may not be accurate, especially when temperatures are changing fast, as they generally do near the beginning of a run.

3.0 INPUT COMMANDS

The explicit method previously available in FLAC has been extended to allow the use of an implicit method. The command sequence to use the implicit method is as follows:

```
SET THDT=value
THSOLVE........IMPLICIT
```

The value of THDT is not restricted by stability. The extra keyword on the THSOLVE command switches to an implicit scheme. It is permissible to change between implicit and explicit solution methods at any time during a run.

4.0 USAGE

The advantage of an implicit method is that the timestep is not restricted by numerical stability. The disadvantages are that:

(1) extra memory is required to use this method;

(2) a set of simultaneous equations must be solved at each timestep; and

(3) larger timesteps may introduce inaccuracy.

These disadvantages must be kept in mind when deciding which method to use. They are discussed below.

Memory Requirement

If an attempt is made to use the implicit method for a large problem, an error message will be generated. The only way to avoid this is to run a smaller problem or use the explicit method.

Solving a Set of Equations

The set of equations to be solved at each timestep is solved iteratively. Each iteration of the solution takes about the same length of time as a single step of the explicit method. The number of iterations depends on the timestep chosen and the particular problem being solved, but is always at least 3. Thus, the implicit scheme offers an advantage over the explicit scheme only if the timestep is much larger than that which the explicit scheme would use. On the other hand, the iterative scheme does introduce some restriction on the timestep. In general, a timestep between 100 and 10,000 times that used by the explicit scheme is satisfactory.

The program displays the iteration counter and a measure of convergence (the residual) to the left of the timestep counter while the implicit scheme is running. The user should check that the number of iterations being taken is such that the implicit scheme is indeed more efficient than the explicit scheme. If not, switch to the explicit scheme or change the timestep. This counter will also indicate if the method is not converging. If the residual is increasing with successive iterations, the method is not converging, and a smaller timestep must be used.

Inaccuracy Due to Large Timesteps

In the initial period of a solution, temperatures generally change much faster than later. It therefore is appropriate to use a smaller timestep or, more likely, the explicit method, initially, and then switch to the implicit method with a large timestep later in the solution. Convergence of the solution generally occurs in fewer iterations at later timesteps.

Selecting the Implicit Method

From the above discussion, it can be seen that the implicit method should be used only at late times in the solution, and only if the timestep can be increased significantly over the one used by the explicit scheme.